

文章编号: 2095-2163(2021)11-0034-09

中图分类号: TP391.1

文献标志码: A

基于双编码和 MRT 的实体与关系联合抽取研究

白羽飞, 高建瓴

(贵州大学 大数据与信息工程学院, 贵阳 550025)

摘要: 部分联合学习模型使用同一个编码器对实体识别和关系抽取这两项任务进行编码,但是单个编码器不足以捕获同一空间中两个任务所需的信息,本文通过使用序列编码器获取实体标签,表格编码器获取关系标签的双编码方式改善这一缺陷;与此同时,为了让整个模型更快、更好的收敛到理想效果,在双编码器模型的基础上加入了最小风险训练来优化全局损失函数。与现有的主流模型在 4 个标准数据集上进行对比发现,本文模型相较主流模型在评价指标上均有一定程度的提升。

关键词: 关系抽取; 实体识别; 联合学习; 双编码器; 最小风险训练

Research on joint extraction of entity and relationship based on two encoders and minimal risk training

BAI Yufei, GAO Jianling

(College of Big Data and Information Engineering, Guizhou University, Guiyang 550025, China)

[Abstract] Some of the joint learning model use the same encoder to encode the two tasks of entity recognition and relationship extraction, but a single encoder is not enough to capture the information required by the two tasks in the same space. By using the sequence encoder to obtain the entity tags and the table encoder to obtain the relationship tags, the Two-Encoders can improve this defect. At the same time, in order to make the entire model converge to the desired effect faster and better, minimum risk training is added to the Two-Encoders model to optimize the global loss function. Comparing with the existing mainstream models on the four standard data sets, it is found that the model in this paper has a certain degree of improvement in evaluation indicators compared with the mainstream models.

[Key words] relation extraction; entity recognition; joint learning; Two-Encoders; minimal risk training

0 引言

随着计算机技术的发展和 Web 数据的日益激增,如何从这些海量数据中获取有效的信息成为了当前的研究热点。信息抽取(Information Extraction, IE)作为一种经典的自然语言处理任务,其主要目标是从半结构化或非结构化的数据中提取实体、关系和事件等结构化的语义信息,是构建结构化知识库和知识图谱的核心部分。信息抽取中的命名实体识别(Named Entity Recognition, NER)和关系抽取(Relation Extraction, RE)同样也广泛应用在知识图谱、问答系统、自动摘要生成等领域,其中关系抽取任务一般建立在给定的已知实体对和文本描述基础上,挖掘实体对之间的语义关系,形成由实体、关系、实体组成的三元组,例如(Los Angeles, Located in, America)。

自然语言处理(Natural Language Processing,

NLP)致力于使用自然语言与计算机进行有效通信,而在神经网络未被广泛关注之前,NLP 任务常常通过基于规则和机器学习的方法来实现。2012 年, Hinton 等人通过 CNN 神经网络构建的 AlexNet 模型在 ImageNet 图像识别比赛中夺得冠军,引起了学者们对于神经网络的广泛关注。随着 GPU 性能的提升,以及 Pytorch 和 TensorFlow 等一批优秀的深度学习框架的出现,越来越多的学者也尝试使用 CNN、Attention、LSTM 和 Transformer 等神经网络模型来处理 NLP 任务。

基于深度学习的实体与关系抽取方法通常可以分为两类:流水线(Pipeline)方法和联合学习(Joint Learning)方法。在联合学习没有被提出前,基于流水线方法的实体与关系抽取模型被广泛使用。其中,基于 RNN、CNN、LSTM 以及其改进模型的网络结构,因灵活简单的特性成为了关系抽取研究者的首选。基于 RNN 模型进行实体关系抽取的方法由

作者简介: 白羽飞(1996-),男,硕士研究生,主要研究方向:深度学习、自然语言处理;高建瓴(1969-),女,硕士,副教授,硕士生导师,主要研究方向:数据库系统、数据挖掘。

通讯作者: 高建瓴 Email:454965711@qq.com

收稿日期: 2021-09-08

Socher 等人于 2012 年首次提出^[1]; Hashimoto 等人于 2013 年提出了基于句法树的递归神经网络 (RNN) 模型^[2]; Zeng 等人于 2014 年首次提出了使用 CNN 进行关系抽取^[3]; Xu 等人于 2015 年在 Zeng 等人工作的基础上提出了基于依存分析树的卷积神经网络的实体关系抽取^[4]; Yan 等人于 2015 年提出了基于 LSTM 的融合句法依存分析树的最短路径, 以及词向量特征、词性特征、WordNet 特征、句法类型特征来进行关系抽取^[5]。然而, 流水线方法割裂了实体识别和关系抽取之间的联系, 所以从模型角度来看, 流水线方法是存在缺陷的。

针对流水线方法存在的问题, 学者们提出了联合学习方法, 并且证明了通过联合学习可以有效地缓解错误传播问题。联合学习方法是把两个相关任务放在一起进行联合训练, 这种方法在两个任务之间有着较强的依赖关系时效果尤为显著, 可以同时提高两个任务的性能。Katiyar 等人于 2017 年首次将注意力机制与双向 LSTM 一起用于联合提取实体和分类关系, 是真正意义上的第一个神经网络联合抽取模型^[6]; Lee 等人于 2019 年开发了端到端循环神经模型用于关系抽取, 该模型包含了实体感知的注意机制和潜在实体类型^[7]。

由于远程监督能借助外部知识库为数据提供标签, 从而省去人工标注的麻烦, 所以部分学者也在基于远程监督的实体与关系联合学习方面开展了一些相关的工作。例如, Vashishth 等人于 2018 年提出了一种基于远程监督的神经网络关系抽取方法 RESIDE^[8]; 吴等人于 2019 年提出了一种带有神经噪声转换器的方法来减轻噪声数据的影响, 并提出一种条件最优选择器来进行正确的预测^[9]。另外, 由于图神经网络在关系抽取任务上的优秀表现, 任亚峰等人于 2021 年提出了一种新的跨图神经模型, 用于联合提取生物学文本中的重叠实体关系, 这种模型分别通过跨度评分器和关系评分器捕获相关实体之间的关系, 最终输出所有有效的关系三元组^[10]。

表格结构的识别是指从表格区域中提取结构化信息, 然后可以将其用于进一步的应用。在表格结构的应用方面, Kostić 和 Risch 在 2021 年提出通过将文本、表格和问题联合编码到单个向量空间来检索与问题相关的文本和表格, 提高了检索性能^[11]。理解表格区域的结构识别是文档解析的一个重要组成部分, Pegu 和 Singh 在 2021 年提出了一种由行识别和列识别模块组成的新型表结构识别模型来解决自动化文档分析和解析^[12]。

序列编码在结构上类似于 Transformer, Yu 和 Wang 在 2021 年发表的论文中构建了 4 组序列标注模型对单一文本信息进行提取, 最高 F1 值达到了 0.994 73^[13]; Wang 和 Xiong 等人于 2020 年提出了一种多目标依赖关系抽取模型, 把关系抽取改造为带有序列标签的任务^[14]。从句子中提取重叠和多重关系的三元组仍然是一个巨大的挑战, 针对这个问题, Zheng 和 Wang 等人于 2021 年提出了一种序列到序列的方法, 其包括一个加权的相对位置 Transformer 编码器来灵活地捕捉实体之间的语义关系^[15]。

为了探究实体与关系之间联系更加紧密的模型, 并且在评价指标上获得领先的分数, 本文提出了一种较为新颖的模型, 该模型由两种相互连接的编码器组成, 并且引入了最小风险训练来优化全局损失函数, 如图 1 所示。表格编码器用于获取关系信息, 序列编码器用于获取实体信息, 并统称为表格-序列编码器或者双编码器。双编码器任意一层的详细信息, 以及其如何进行交互, 如图 2 所示。

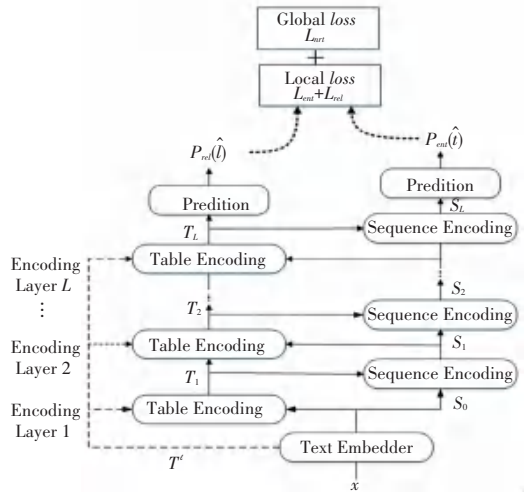


图 1 结合双编码器与最小风险训练的模型概述图
Fig. 1 Overview of the model combining Two-Encoders and minimum risk training

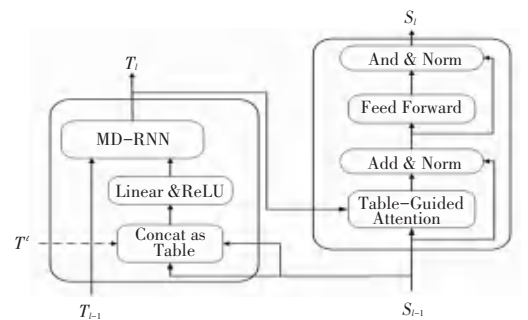


图 2 表格-序列编码器中的某一层
Fig. 2 A layer in Table-Sequence encoder

1 表格编码器

1.1 表格编码

在众多实体与关系联合抽取的方法中,较为新颖且流行的方法是表格编码。2014年, Miwa 和 Sasaki 最先使用表格填充的方法对命名实体识别和关系抽取进行联合探究,使得联合抽取简化为表格填充这样一个足够灵活的任务,并且提出先实体后关系、先近距离关系再远距离关系的搜索顺序,以及从实体到关系的全局特征这样两个研究方案^[16]。但是其使用的方法是基于历史的结构化预测,随着基于神经网络的深度学习逐渐流行,部分学者开始利用基于深度学习的表格编码来解决实体与关系联合抽取问题。2016年, Gupta 和 Schütze 提出表格填充的多任务循环神经网络(TF-MTRNN)模型,与 Miwa 和 Sasaki 提出的模型相比较,TF-MTRNN 模型利用神经网络强大的自学习和特征提取能力,在实体识别和关系抽取这两个任务上 F1 值分别提高了 2.0% 和 2.7%^[17]。

在图3所示包含实体和关系的句例中,被训练的句子中实体和关系都已被打上相应的标签,其中“PER”表示“人物”实体,“LOC”表示“地点”实体,并且实体之间的关系是有向关系。图4展示了图3中句子与对应的实体关系表,可以看出表格是正方形,所以含有 n 个单词的句子构建的表格中单元格数为 $n \times n$ 。表格内容沿着左上至右下对角线对称,且该对角线的第 k 个单元格对应输入句子的第 k ($1 \leq k \leq n$) 个单词的实体标签,其中第 i 行和第 j 列的单元格对应输入句子的第 i 个和第 j 个单词之间的关系标签($i \neq j; 1 \leq i, j \leq n$)。关系标签由关系类型和方向组成,“上”表示非关系对,“ \rightarrow ”和“ \leftarrow ”分别表示从左至右以及从右至左关系。本文采用的是 BILOU (Beginning, Inside, Last, Outside, Unit) 序列标注方法,该方法已被证明优于传统的 BIO (Beginning, Inside, Outside) 方法^[18]。

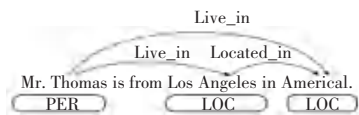


图3 包含实体和关系的句例

Fig. 3 A sentence example containing entities and relationships

图4与图5做比较可得出,本文采用的表格填充方式并没有完全遵循 Miwa 和 Sasaki 的论文,不同

点在于:本文填充整个表而不是只填充较低(或较高)的三角部分;本文将关系标签分配给实体跨越交叉的所有单元格,而不是实体最后一个单词交叉的单元格。

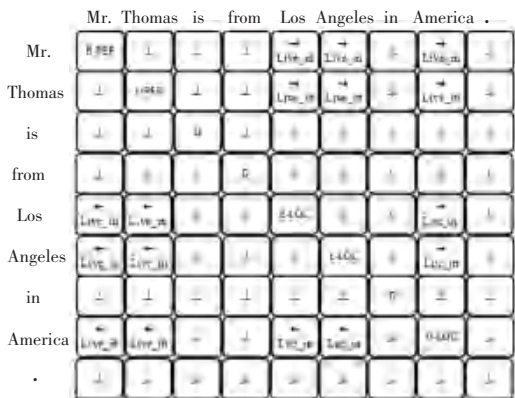


图4 表格填充表示

Fig. 4 Table filling representation

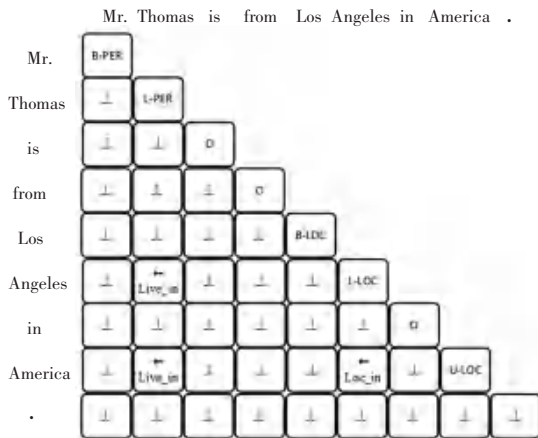


图5 Miwa 和 Sasaki 提出的表格填充表示

Fig. 5 Table filling representation proposed by Miwa and Sasaki

1.2 多向多维循环神经网络(MD-MDRNNs)

多维循环神经网络 MDRNNs (Multi-Dimensional Recurrent Neural Networks) 的基本思想是用与数据维度一样多的循环连接来取代标准 RNN 中的单个循环连接。对于输入序列的点 (x_1, x_2, \dots, x_n) , 利用 MDRNNs 可以访问所有点 $(x'_1, x'_2, \dots, x'_n)$, 其中 $x'_k \leq x_k$ 对 $\forall k \in (1, \dots, n)$ 都成立。例如:二维序列点 (x'_i, x'_j) 就属于图6中 context region 部分,在没有多向的前提下,一个完整序列的二维上下文区域就只能是 context region 部分。然而,在应用需求上更希望网络能够全方位的接触到周围的环境,这对于需要精确定位的任务尤其如此,例如图像分割、目标检测、命名实体识别以及关系抽取等任务。

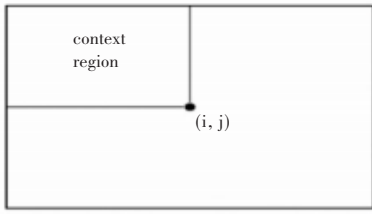


图 6 单向 2DRNN 模型在 (i, j) 点处可访问的上下文区域

Fig. 6 The context area accessible by the unidirectional 2DRNN model at point (i, j)

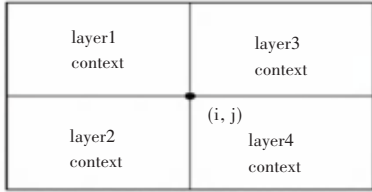


图 7 多向 2DRNNs 在 (i, j) 处可访问的上下文区域

Fig. 7 The context area accessible by the multiple directions 2DRNN model at point (i, j)

1997 年, Schusterd 和 Paliwal 提出的双向循环神经网络(BRNNs)解决了一维 RNN 的多向上下文问题, BRNNs 包含连接到同一个输出层的两个独立隐藏层, 其分别从正向和反向处理输入序列, 从而为网络提供上下文访问。四向 2DRNNs 情况如图 8 所示, 通过不同的轴选择搭配获得了 4 个前向传播方向。而从图 7 可以看出, 一个隐藏层代表一种传播方法, 最后连接成一个可以访问周围所有上下文的输出层。

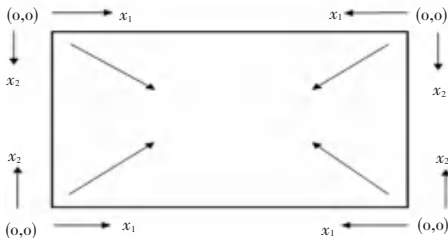


图 8 多向 2DRNNs 传播情况

Fig. 8 Propagation of multi-directional 2DRNNs

对于表格编码器中神经网络的隐藏层, 可以使用包含 4 个方向的 3DRNNs 来计算其隐藏状态。一个隐藏层代表一种传播方法, 最后连接成一个可以访问周围所有上下文的输出层, 如图 9 所示, 使用 D+或 D-来指示隐藏状态在 D 维单元格之间流动的方向, 其中 D 可以是层、行或列, 而隐藏层之间的传播方向往往是从低层向高层流动, 所以层维的流动方向固定为 layer⁺。图 9 中省略了 (b)、(c) 和 (d) 的低层 T_{l-1} 和输入 X_{l,i,j}, 因为其与 (a) 相同。表格编码器原理: 首先, 将每一层序列表示的输出进行两两组合, 即以 (i, j) 为坐标, (S_{l-1,i}; S_{l-1,j}) 为值来构

造一个非上下文化的表格; 其次, 用一个完全连接层将隐藏层的大小减半, 使得 (S_{l-1,i}; S_{l-1,j}) 变为 X_{l,i,j}, 如公式 (1) 所示。最后, 使用带有门控循环单元(Gated Recurrent Unit, GRU)的多向多维循环神经网络(MD-MDRNNs)来上下文化 X_l, 这样就能通过迭代的方式计算每个单元的隐藏状态, 并形成具有上下文化的表格表示 T_l, 如公式 (2) 所示。

$$X_{l,i,j} = ReLU(Linear([S_{l-1,i}; S_{l-1,j}])) \quad (1)$$

$$T_{l,i,j} = GRU(X_{l,i,j}, T_{l-1,i,j}, T_{l,i-1,j}, T_{l,i,j-1}) \quad (2)$$

由公式 (2) 可以看出, 表格编码器利用了层、行和列这 3 个维度的上下文信息, 也就是说, 其不仅考虑相邻行和列的单元格 T_{l,i-1,j} 与 T_{l,i,j-1}, 还考虑前一层 T_{l-1,i,j}。另外, 通过多次实验发现仅考虑图 9 中 (a) 和 (c) 方向并不比考虑 4 个方向获得的性能差。因此, 为了减少计算量, 只设置 (a) 和 (c) 方向作为默认方向, 且最终多向多维的表格表示 T_{l,i,j} 是这两个 RNN 隐藏状态的连接, 如公式 (5) 所示。其中, 公式 (3) 为 (a) 方向的表格表示, 公式 (4) 为 (c) 方向的表格表示。

$$T_{l,i,j}^{(a)} = GRU^{(a)}(X_{l,i,j}, T_{l-1,i,j}^{(a)}, T_{l,i-1,j}^{(a)}, T_{l,i,j-1}^{(a)}) \quad (3)$$

$$T_{l,i,j}^{(c)} = GRU^{(c)}(X_{l,i,j}, T_{l-1,i,j}^{(c)}, T_{l,i+1,j}^{(c)}, T_{l,i,j+1}^{(c)}) \quad (4)$$

$$T_{l,i,j} = [T_{l,i,j}^{(a)}; T_{l,i,j}^{(c)}] \quad (5)$$

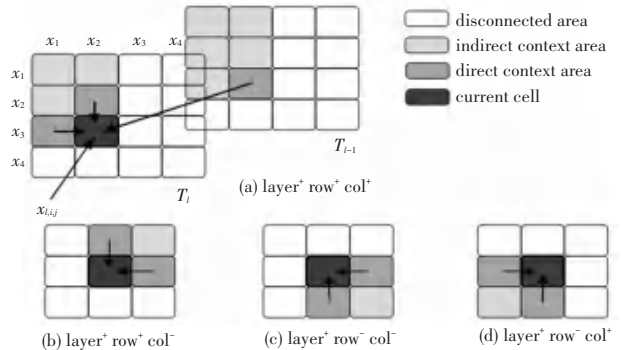


图 9 包含 4 个方向的 3DRNNs 示意图

Fig. 9 Schematic diagram of 3DRNNs with four directions

2 序列编码器

序列编码器用于学习向量的序列表示, 其中第 i 个向量对应于输入句子的第 i 个单词, 该架构类似于 Transformer, 具体组成如图 2 的右边部分所示^[19]。值得注意的是, 本文使用表格引导的注意力取代了规模的点积注意力。注意力函数的本质可以描述为一个查询 Q_i 到一系列键值对(key-value)的映射, 而计算 attention 值主要分为 3 步, 第一步是将需要查询的 Q_i 和每个 key 进行相似度计算得到权重, 第二步通常是使用一个 softmax 函数对这些权

重进行归一化,最后一步将权重与相应的 *value* 进行加权求和得到最后的 *attention*。注意力机制的一般形式如图 10 所示,给定查询 Q (queries)、键 K (keys)、值 V (values),对于每个查询 Q_i ,其对应的输出 O_i 是所有 *values* 的加权和,而分配给每个 *value* 的权重由这个查询 Q_i 与所有 *keys* 的相关性决定,其中相关性由得分函数 f 给出。根据每个查询 Q_i 和键 K_j ,Bahdanau 等人定义了得分函数 $f(Q_i, K_j) = U \cdot g(Q_i, K_j)$, U 是一个可学习向量, $g(Q_i, K_j) = \tanh(Q_i W_0 + K_j W_1)$ 是将每个键值对和查询映射到一个向量的函数, W_0 、 W_1 是可学习参数。

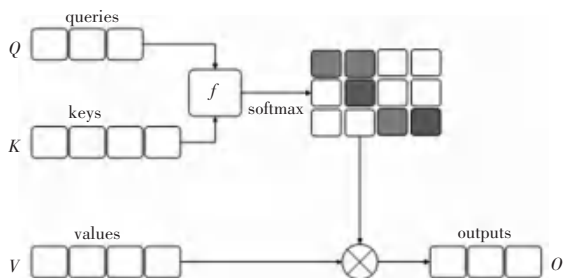


图 10 注意力机制的一般形式

Fig. 10 The general form of the attention mechanism

注意力机制本质上是一种自注意机制,其中查询、键和值完全相同,且本质上是前一层 S_{l-1} 的序列表示,即 $Q = K = V = S_{l-1}$ 。从图 1 中序列表示 S_{l-1} 会作为表格表示 T_l 的一部分输入,因此可以认为 T_l 也是 Q_i 和 K_j 的函数,使得 $T_{l,i,j} = g(S_{l-1,i}, S_{l-1,j}) = g(Q_i, K_j)$ 与 $f(Q_i, K_j) = U \cdot g(Q_i, K_j)$ 整合得到表格引导式的注意力机制的得分函数为式(6):

$$f(Q_i, K_j) = U \cdot T_{l,i,j} \quad (6)$$

使用这种表格引导式的注意力机制有以下 3 个优点:

- (1) 不需要计算 g 函数,因为 T_l 已经从表格编码器获得;
- (2) 经过行、列和层这 3 个维度上下文化的 T_l , 分别对应前一层中的查询、键、键与查询,这样的上下文信息可以让网络更好地捕获更困难的单词与单词之间的依赖关系;
- (3) 允许表格编码器参与到序列表示的学习过程,从而形成两个编码器之间的双向交互。

3 双编码器训练过程

利用公式(7)和(8)预测实体标签和关系标签的概率分布。

$$P_{ent}(\hat{t} | s; \theta_E) = \text{Softmax}(\text{Linear}(S_{L,i})) \quad (7)$$

$$P_{rel}(\hat{l} | s, e_i, e_j; \theta_R) = \text{softmax}(\text{Linear}(T_{L,i,j})) \quad (8)$$

其中, t_i 为第 i 个单词的标签; S_L 和 T_L 分别对应序列表示和表格表示; P_θ 是概率估计函数; θ 是模型参数。

在训练阶段,NER 和 RE 均采用常用的交叉熵损失来计算损失函数,给定输入文本 x 及其正确的序列标签 y^{NER} 和表格标签 y^{RE} ,通过公式(9)和(10)分别计算 NER 和 RE 的损失函数。

$$L_{ent} = - \frac{1}{|s|} \sum_{i=1}^{|s|} \log P_{ent}(\hat{t}_i = t_i | s; \theta_E) \quad (9)$$

$$L_{rel} = - \sum_{\substack{e_i, e_j \in \hat{\mathcal{E}} \\ e_i \neq e_j}} \frac{\log P_{rel}(\hat{l} = l | s, e_i, e_j; \theta_R)}{|\hat{\mathcal{E}}| (|\hat{\mathcal{E}}| - 1)} \quad (10)$$

其中, $\hat{\mathcal{E}}$ 表示一组检测到的实体; \hat{t} 为实体标签序列;而候选实体对 (e_i, e_j) 的真实标签 l 可以从真实注释中读取; s 表示输入的句子。

训练的最终目标是要损失和 $\tilde{L} = L_{ent} + L_{rel}$ 最小。在评估过程中,关系的预测依赖于实体的预测,因此首先对实体进行预测,然后再查找关系概率表 $P_\theta(y^{RE})$,看看预测的实体之间是否存在有效的关系。

4 联合最小风险训练

$\tilde{L} = L_{ent} + L_{rel}$ 是一种局部损失函数,原因在于 L_{ent} 和 L_{rel} 都是计算局部的损失函数。例如,损失函数 L_{ent} 是基于局部实体标签 t_i 的正确性,而不是基于全局指标,如实体模块的 F1 值。另外,由于 L_{ent} 和 L_{rel} 是相互独立的,这就导致实体模块需要等待关系模块更新共享参数之后才能继续训练,而不是直接从关系模块的损失函数中得到反馈监督。

为了加强实体模块与关系模块之间的交互,提高联合模型的整体性能,可以在双编码器模型的基础上加入最小风险训练 (MRT, Minimum Risk Training)。MRT 提供了一种轻量级的方法来加强实体模块和关系模块之间的联系,同时又保持其各自的功能不受影响。具体地说,给定输入 x 和损失函数 $\Delta(\hat{y}, y)$, MRT 的目标是寻找一个后验概率分布 $P(\hat{y} | x)$ 来最小化期望损失 $E_{\hat{y} \sim P(\hat{y} | x)} \Delta(\hat{y}, y)$ 。引入 MRT 到联合模型的优点在于, MRT 可以显式地优化全局语句级损失函数,比如 F1 值。因此,可以在训练阶段捕获更多的句子级信息,同时在测试阶段可以更好地匹配评价指标。与以往的单任务

MRT 相比, 本文的联合 MRT 算法会将不同子模块的消息进行整合, 这也是联合学习中增强实体模块与关系模块交互的关键步骤。因此, 实体模型在训练阶段可以直接使用关系模型的损失, 而无需等待共享参数, 反之亦然。

在利用 MRT 求全局损失函数之前, 先定义联合概率分布, 如公式(11)所示。

$$P(\hat{y} | s; \theta) = P(\hat{\varepsilon} | s; \theta_E) P(\hat{R} | s, \hat{\varepsilon}; \theta_R) = \prod_i P_{ent}(\hat{t}_i | s; \theta_E) \prod_{\substack{e_i, e_j \in \hat{\varepsilon} \\ e_i \neq e_j}} P_{rel}(\hat{l} | s, e_i, e_j; \theta_R) \quad (11)$$

其中, 集合 $y = (\varepsilon, R)$ 包含正确的实体标签和关系标签; $\hat{y} = (\hat{\varepsilon}, \hat{R})$ 表示联合学习模型的输出; 而 $\theta = \theta_E \cup \theta_R$ 是联合模型的参数; $y(s)$ 是输入句子 $s(y, \hat{y} \in y(s))$ 的所有可能的输出集合。

MRT 的目标是使公式(12)所示的期望损失最小。

$$E_{\hat{y} \sim P(\hat{y} | s; \theta)} \Delta(\hat{y}, y) = \sum_{\hat{y} \in y(s)} P(\hat{y} | s; \theta) \Delta(\hat{y}, y) \quad (12)$$

其中, 损失函数 $\Delta(\hat{y}, y)$ 描述 \hat{y} 和 y 的差异程度, 是提高联合模型性能的关键因素。

本文考虑句子级别的实体识别性能和关系抽取性能, 并把其分别表示为 $F_{ent}(\hat{\varepsilon}, \varepsilon)$ 和 $F_{rel}(\hat{R}, R)$, 且用 $1 - F_{ent}(\hat{\varepsilon}, \varepsilon)$ 和 $1 - F_{rel}(\hat{R}, R)$ 作为实体损失和关系损失。这样设置的原因: 一方面 $F1$ 值表征了输出的整体性能, 并使训练阶段和测试阶段的评估指标一致; 另一方面, $F1$ 值不会像局部损失函数 L_{ent} 和 L_{rel} 那样分解为 $\hat{\varepsilon}$ 和 \hat{R} 的局部预测。

利用联合 MRT 优化两个任务的损失, 不仅可以让实体模型根据关系模型的损失函数预测候选实体的可信程度, 还可以让关系模型获得实体识别结果的合理程度。因此, 可以通过相加的方式定义全局损失函数, 如公式(13)所示。

$$\Delta_{E+R}(\hat{y}, y) = 1 - \frac{1}{2} [F_{ent}(\hat{\varepsilon}, \varepsilon) + F_{rel}(\hat{R}, R)] \quad (13)$$

为了训练联合学习模型, 本文首先通过最小化局部损失函数 \bar{L} 来对模型进行预训练, 然后再最小化局部和全局损失函数之和 $\bar{L} + L_{mrt}$ 。这个设置与其他学者对于 MRT 的应用的不同在于其在第二阶段只优化 L_{mrt} , 而本文通过实验发现在第二阶段中加入 \bar{L} , 会使得训练过程更加稳定。

5 实验

5.1 数据集及评价指标

实验在 ACE04、ACE05、CoNLL04 和 ADE 4 个

数据集上使用准确率 (*Precision*)、召回率 (*Recall*) 和 $F1$ 值作为实体识别和关系抽取的评价指标, 对本文模型进行评估。对于命名实体识别任务, 当且仅当实体的类型和边界都与真实值匹配时, 实体预测才是正确的。同样, 对于关系抽取任务, 只有当两个实体的关系类型和关系指向都与真实值匹配时, 关系预测才被认为是正确的。另外还增加了严格关系 RE+, 当关系的类型和指向以及实体的边界和类型都与标准数据匹配时, 则认为该严格关系预测是正确的。

5.2 结果与分析

近几年一些主流关系抽取模型和本文模型分别在 ACE04、ACE05、CoNLL04 和 ADE 这 4 个数据集上的 NER、RE 以及 RE+ 的 $F1$ 值如表 1~4 所示, 其中 ‘-’ 表示此项无值。表 1~表 4 中数据的柱状图呈现如图 11~14 所示, 可以看出本文提出的模型在 NER、RE 和 RE+ 的 $F1$ 值均比其他模型高。

表 1 各类主流模型在 ACE04 中的 $F1$ 值

Tab. 1 $F1$ value of various mainstream models in ACE04

Model	NER	RE	RE+
Li and Ji (2014)	79.7	48.3	45.3
Katiyar and Cardie (2017)	79.6	49.3	45.7
Bekoulis et al. (2018b)	81.2	-	47.1
Bekoulis et al. (2018a)	81.6	-	47.5
Miwa and Bansal (2016)	81.8	-	48.4
Li et al. (2019)	83.6	-	49.4
Luan et al. (2019)	87.4	59.7	-
Wang and Lu (2020)	88.6	63.3	59.6
Ours	90.8	66.7	62.6

表 2 各类主流模型在 ACE05 中的 $F1$ 值

Tab. 2 $F1$ value of various mainstream models in ACE05

Model	NER	RE	RE+
Li and Ji (2014)	80.8	52.1	49.5
Miwa and Bansal (2016)	83.4	-	55.6
Katiyar and Cardie (2017)	82.6	55.9	53.6
Zhang et al. (2017)	83.6	-	57.5
Sun et al. (2018)	83.6	-	59.6
Li et al. (2019)	84.8	-	60.2
Dixit and Al (2019)	86.0	62.8	-
Luan et al. (2019)	88.4	63.2	-
Wadden et al. (2019)	88.6	63.4	-
Wang and Lu (2020)	89.5	67.3	63.9
Ours	90.7	69.5	65.7

表 3 各类主流模型在 CoNLL04 中的 F1 值

Tab. 3 F1 value of various mainstream models in CoNLL04

Model	NER	RE	RE+
Miwa and Sasaki (2014)	80.7	-	61.0
Bekoulis et al. (2018a)	83.6	-	62.0
Bekoulis et al. (2018b)	83.9	-	62.0
Tran and Kavuluru (2019)	84.2	-	62.3
Nguyen and Verspoor (2019)	86.2	-	64.4
Zhang et al. (2017)	85.6	-	67.8
Li et al. (2019)	87.8	-	68.9
Eberts and Ulges (2019)	86.3	-	70.5
Wang and Lu (2020)	88.1	73.0	71.8
Ours	89.7	74.6	73.6

表 4 各类主流模型在 ADE 中的 F1 值

Tab. 4 F1 value of various mainstream models in ADE

Model	NER	RE	RE+
Liet al. (2016)	79.5	-	63.4
Liet al. (2017)	84.6	-	71.4
Bekoulis et al. (2018b)	86.4	-	74.6
Bekoulis et al. (2018a)	86.7	-	75.5
Tran and Kavuluru (2019)	87.1	-	77.3
Eberts and Ulges (2019)	89.3	-	78.9
Wang and Lu (2020)	89.7	80.1	80.1
Ours	91.2	82.4	82.4

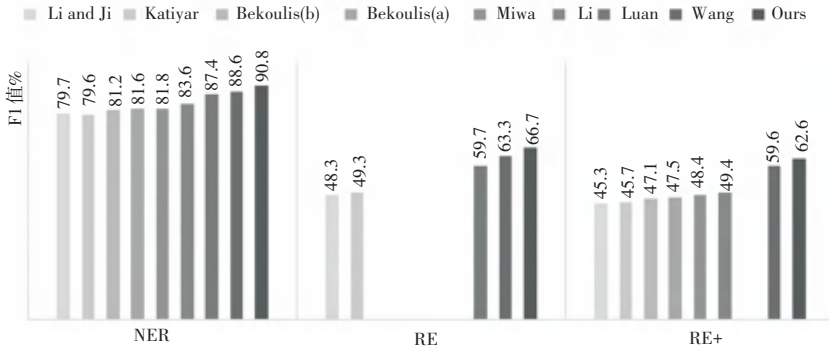


图 11 各类主流模型在 ACE04 中的 F1 值柱状图

Fig.11 F1 value histogram of various mainstream models in ACE04

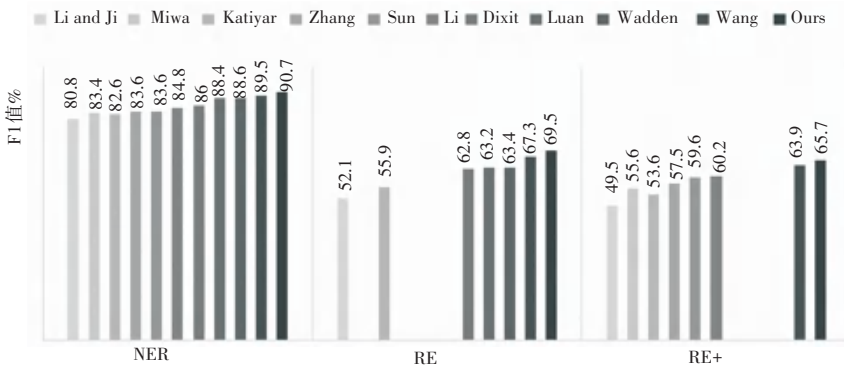


图 12 各类主流模型在 ACE05 中的 F1 值柱状图

Fig. 12 F1 value histogram of various mainstream models in ACE05

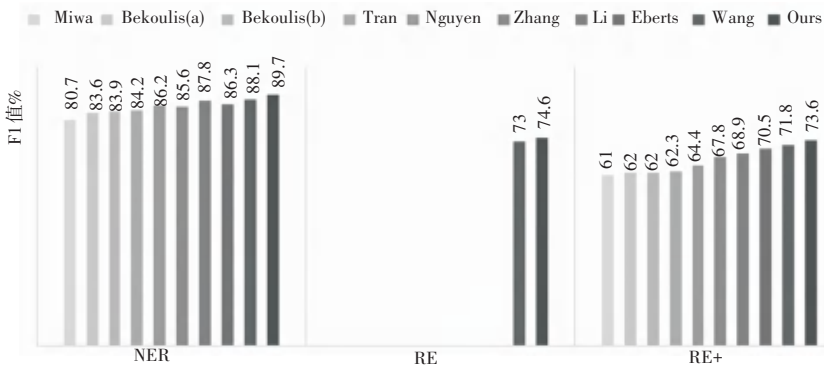


图 13 各类主流模型在 CoNLL04 中的 F1 值柱状图

Fig. 13 F1 value histogram of various mainstream models in CoNLL04

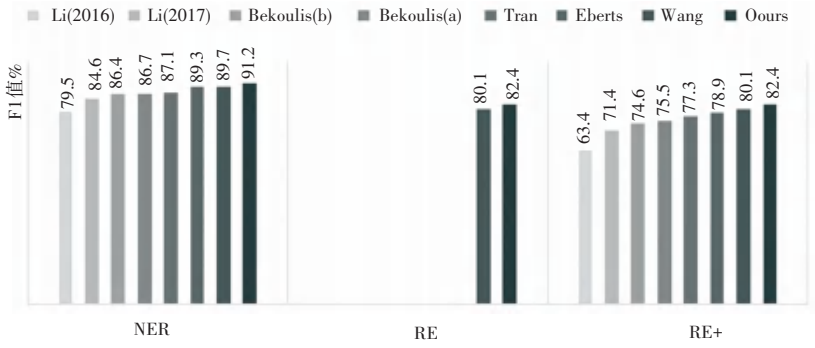


图 14 各类主流模型在 ADE 中的 F1 值柱状图

Fig. 14 F1 value histogram of various mainstream models in ADE

表 1~表 4 中 NER、RE 和 RE+ 的准确率 P 、召回率 R 以及 $F1$ 值的展开, 见表 5~8。若把所有模型的数据全部展开会显得繁杂且不容易突出重点信息, 所以表 5~8 只选取了近两年的模型数据进行展开。

表 5 部分模型在 ACE04 中的 P 、 R 、 $F1$ 值

Tab. 5 P , R , $F1$ values of some models in ACE04

Model	NER			RE			RE+		
	P	R	$F1$	P	R	$F1$	P	R	$F1$
Bekoulis et al. (2018b)	81.0	81.3	81.2	-	-	-	50.1	44.5	47.1
Li et al. (2019)	84.4	82.9	83.6	-	-	-	50.1	48.7	49.4
Wang and Lu (2020)	89.2	88.0	88.6	66.2	60.8	63.3	63.0	58.1	59.6
Ours	91.0	90.5	90.8	71.1	62.6	66.7	67.1	59.0	62.6

表 6 部分模型在 ACE05 中的 P 、 R 、 $F1$ 值

Tab. 6 P , R , $F1$ values of some models in ACE05

Model	NER			RE			RE+		
	P	R	$F1$	P	R	$F1$	P	R	$F1$
Li et al. (2019)	84.7	84.9	84.8	-	-	-	64.8	56.2	60.2
Dixit and Al (2019)	85.9	86.1	86.0	68.0	58.4	62.8	-	-	-
Wang and Lu (2020)	89.8	89.2	89.5	76.4	59.6	67.3	71.8	56.5	63.9
Ours	90.2	91.1	90.7	78.2	62.5	69.5	73.6	58.4	65.7

表 7 部分模型在 CoNLL04 中的 P 、 R 、 $F1$ 值

Tab. 7 P , R , $F1$ values of some models in CoNLL04

Model	NER			RE			RE+		
	P	R	$F1$	P	R	$F1$	P	R	$F1$
Bekoulis et al. (2018b)	83.8	84.1	83.9	-	-	-	63.8	60.4	62.0
Li et al. (2019)	89.0	86.6	87.8	-	-	-	69.2	68.2	68.9
Wang and Lu (2020)	89.1	87.1	88.1	76.8	69.7	73.0	75.5	68.5	71.8
Ours	90.8	88.9	89.7	79.7	69.4	74.6	79.0	68.5	73.6

表 8 部分模型在 ADE 中的 P 、 R 、 $F1$ 值

Tab. 8 P , R , $F1$ values of some models in ADE

Model	NER			RE			RE+		
	P	R	$F1$	P	R	$F1$	P	R	$F1$
Bekoulis et al. (2018b)	84.7	88.2	86.4	-	-	-	72.1	77.2	74.6
Eberts and Ulges (2019)	89.0	89.6	89.3	-	-	-	77.8	89.0	78.9
Wang and Lu (2020)	88.9	90.7	89.7	79.0	82.0	80.1	79.0	82.0	80.1
Ours	90.9	91.4	91.2	82.6	82.2	82.4	82.6	82.2	82.4

6 结束语

本文通过实验证实了在学习过程中对实体模块使用序列编码器,关系模块使用表格编码器,并且两个编码器之间相互传递信息,在彼此独立的基础上又相互依赖,这样对于捕获两种不同类型的信息是有益的。此外,在双编码器模型的基础上还引入了最小风险训练,最小风险训练算法构建了能让实体和关系模块一起进行反向传播的全局损失函数,与只能优化单个模块参数的局部损失函数相比,全局损失函数的加入让本文模型真正做到了实体与关系的联合学习。

参考文献

- [1] SOCHER R, HUVAL B, MANNING C D, et al. Semantic compositionality through recursive matrix - vector spaces [C] // Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning. 2012: 1201-1211.
- [2] HASHIMOTO K, MIWA M, TSURUOKA Y, et al. Simple customization of recursive neural networks for semantic relation classification [C] // Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013: 1372-1376.
- [3] ZENG D, LIU K, LAI S, et al. Relation classification via convolutional deep neural network [C] // Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics; Technical Papers. 2014: 2335-2344.
- [4] XU K, FENG Y, HUANG S, ZHAO D. Semantic relation classification via convolutional neural networks with simple negative sampling [J]. Computer Science, 2015, 71: 941-949.
- [5] XU Y, MOU LL, LI G, et al. Classifying relation via Long Short - Term Memory networks along shortest dependency paths. Conf. on Empirical Methods in Natural Language Processing, 2015, 42 (1): 56-61.
- [6] KATIYAR A, CARDIC C. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees [C] // Proceeding of the Meeting of the Association for Computational Linguistics. 2017: 917-928.
- [7] LEE J, SEO S, CHOI Y S. Semantic relation classification via bidirectional lstm networks with entity-aware attention using latent entity typing [J]. Symmetry, 2019, 11(6): 785.
- [8] VASHISHTH S, JOSHI R, PRAYAGA S S, et al. Reside: Improving distantly supervised neural relation extraction using side information [J]. arXiv preprint arXiv:1812.04361, 2018.
- [9] WU S, FAN K, ZHANG Q. Improving distantly supervised relation extraction with neural noise converter and conditional optimal selector [C] // Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33(1): 7273-7280.
- [10] FEI H, ZHANG Y, REN Y, et al. A span-graph neural model for overlapping entity relation extraction in biomedical texts [J]. Bioinformatics, 2021, 37(11): 1581-1589.
- [11] KOSTI B, RISCH J, MÖLLER T. Multi-modal Retrieval of Tables and Texts Using Tri-encoder Models [J]. arXiv preprint arXiv:2108.04049, 2021.
- [12] PEGU B, SINGH M, AGARWAL A, et al. Table Structure Recognition Using CoDec Encoder - Decoder [C] // International Conference on Document Analysis and Recognition. Springer, Cham, 2021: 66-80.
- [13] YU B, WANG K. Research on Express Information Extraction Based on Multiple Sequence Labeling Models [C] // IOP Conference Series: Earth and Environmental Science. IOP Publishing, 2021, 632(2): 022060.
- [14] WANG L, XIONG C, DENG N. A Research on Overlapping Relationship Extraction Based on Multi-objective Dependency [C] // 2020 15th International Conference on Computer Science & Education (ICCSSE). IEEE, 2020: 618-622.
- [15] ZHENG W, WANG Z, YAO Q, et al. WRTRe: Weighted relative position transformer for joint entity and relation extraction [J]. Neuro computing, 2021, 459: 315-326.
- [16] MIWA M, SASAKI Y. Modeling joint entity and relation extraction with table representation [C] // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014: 1858-1869.
- [17] GUPTA P, SCHÜTZE H, ANDRASSY B. Table filling multi-task recurrent neural network for joint entity and relation extraction [C] // Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics; Technical Papers. 2016: 2537-2547.
- [18] RATINOV L, ROTH D. Design challenges and misconceptions in named entity recognition [C] // Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009). 2009: 147-155.
- [19] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need [C] // Advances in neural information processing systems. 2017: 5998-6008.

(上接第33页)

- [2] 宋亮,万建洲. 缺失数据插补方法的比较研究 [J]. 统计与决策, 2020, 36(18): 10-14.
- [3] 刘佳星,张宏烈,刘艳菊,等. 基于缺失率的不完整数据填补算法 [J]. 统计与决策, 2021, 37(2): 39-41.
- [4] LITTLE R J A, RUBIN D B. Statistical Analysis With Missing Data [M]. New York: Wiley and Sons Inc, 1987: 7-8.
- [5] 岳廷妍,邱建青,裴姣,等. 数据缺失机制识别联合模型及评价 [J]. 统计与决策, 2019, 35(16): 71-74.
- [6] 章涛,朱麟,季加东,等. 基于 R 软件的缺失数据 MICE 填补效果研究 [J]. 中国卫生统计, 2015, 32(4): 580-584.